

# Tree Traversal

## Pre-Order (node - left - right)

traverse-nlr(t-node) {

1. Recursion base case (en uztaki node'lar iain) oocugu yoksa döndür.

2. Oocugu varsa  
traverse kendisini döndür  
sol oocuk icin  $\rightarrow n.leftchild$   
sağ oocuk icin  $\rightarrow n.rightchild$

### Iterationlar

1 → n

1 2 → traverse 1.leftchild (return n)

1 2 3 → traverse 2.leftchild (return n)

1 2 3 4

1 2 3 4 (3'e döndü)

1 2 3 4 (2'ye döndü)

1 2 3 4 5 → traverse 2.rightchild

1 2 3 4 5 6 → traverse 1.rightchild

---

## Inorder Traversal (left - node - right)

traverse-lnr(t-node) {

1. Recursion base case:  
Oocugu yoksa kendisini döndür.

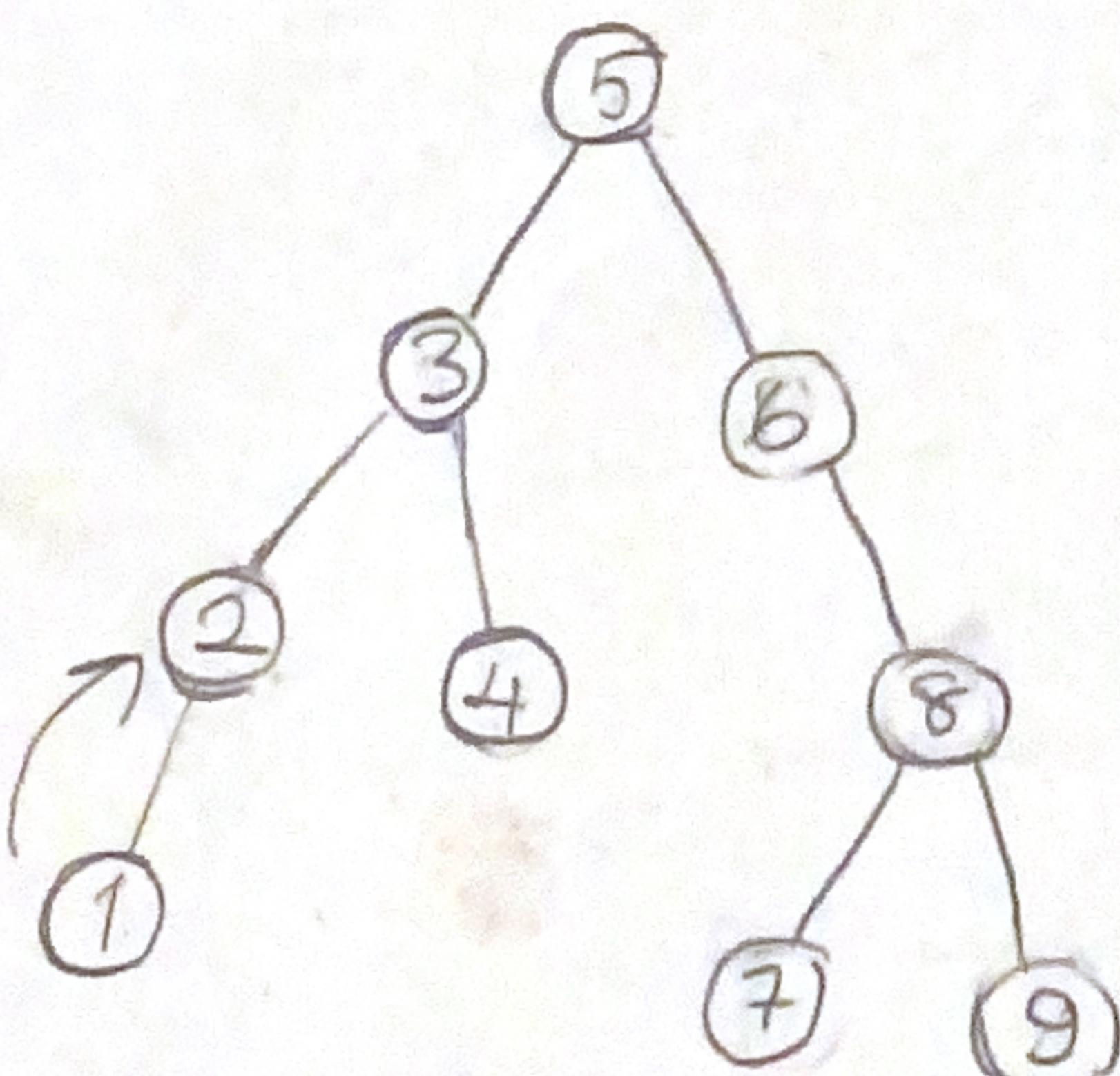
2. Oocugu varsa;

traverse-lnr(sol oocuk)

visit-tree(t-node)

traverse-lnr(sağ oocuk) }

1



## Post Order Traversal (left-right-node)

traverse-lrn(t-node) {

  gocugu yoksa value'yu  
  döndür (yaprak node)  
  gocugu varsa {

    traverse-lrn(t-node.lchild)  
    (sol gocugu traverse et)  
    traverse-lrn(t-node.rchild)  
    visittree(t-node)  
    (kendisini döndür)  
  }

